

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-053224

(43)Date of publication of application : 26.02.1999

(51)Int.Cl.

G06F 11/30

(21)Application number : 09-204625

(71)Applicant : DENSO CORP

(22)Date of filing : 30.07.1997

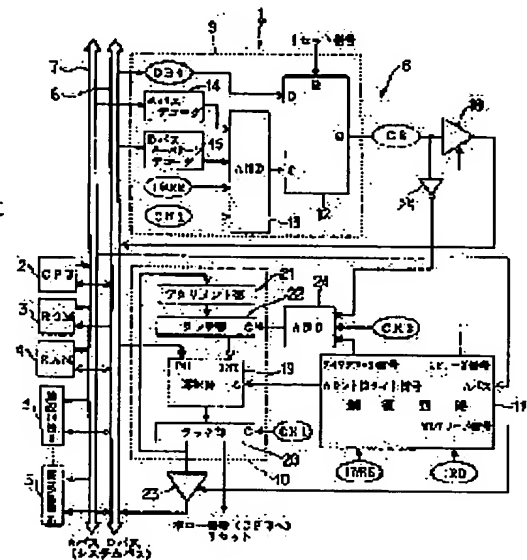
(72)Inventor : AZUMA HIDEJI

## (54) WATCH DOG TIMER AND MICROCOMPUTER

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To easily provide debug of program development by simply permitting the operation and stop of a watch dog timer without performing troublesome connection.

**SOLUTION:** A CPU 2 is connected through a ROM 3, a RAM 4, a peripheral circuit 5, a watch dog timer 6 and a system bus and writes a count initial value to the timer 6 in every prescribed cycle, operating according to a program stored in the ROM 3. The timer 6 reduces a count value one by one in every prescribe cycle when the count initial value is written from the CPU 2 and outputs a borrow signal to the CPU 2 when the value becomes zero. The CPU 2 accesses a CE generation circuit 9 of the timer 6 when it stops the timer 6 at the time of emulation. Because the circuit 9 outputs a CE signal with that, the counter function of a down counter 10 is stopped.





## 【特許請求の範囲】

【請求項 1】 CPU からアクセスされる毎に初期値或いは任意の設定値からカウント動作を実行するように設けられ、限度値までのカウント動作が終了したときは上記 CPU へ異常信号を出力すると共に停止指令を受けた状態でカウント動作を停止するカウント手段と、前記 CPU により第 1 のアクセス条件でアクセスされたときは前記カウント手段へ停止指令を出力すると共に第 2 のアクセス条件でアクセスされたときは停止指令の出力を停止する停止指令出力手段とを備えたことを特徴とするウォッチドッグタイマ。

【請求項 2】 前記停止指令出力手段は、前記 CPU により自己のアドレスに対して第 1 のデータの書き込み動作が行われたときに停止指令を出力すると共に第 2 のデータの書き込み動作が行われたときに停止指令の出力を停止することを特徴とする請求項 1 記載のウォッチドッグタイマ。

【請求項 3】 外部からの指令に応じて前記停止指令出力手段による停止指令の出力状態を出力する出力状態出力手段を備えたことを特徴とする請求項 1 または 2 記載のウォッチドッグタイマ。

【請求項 4】 外部からの指令に応じて前記カウント手段のカウント値を出力するカウント値出力手段を備えたことを特徴とする請求項 1 乃至 3 の何れかに記載のウォッチドッグタイマ。

【請求項 5】 請求項 1 乃至 4 の何れかのウォッチドッグタイマを備えたことを特徴とするマイクロコンピュータ。

## 【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、CPU が暴走したときに当該 CPU をリセットしたり或いは割込みをかけるためのウォッチドッグタイマ及び斯様なウォッチドッグタイマを備えたマイクロコンピュータに関する。

【0002】

【従来の技術】 従来より、ウォッチドッグタイマとして、特開平 3-194628 号公報のものが提案されている。このものは、エミュレーション時に、特別に設けられた外部端子の信号レベルをハイレベルとすることによりウォッチドッグタイマを停止するように構成されており、その外部端子に切替えるための信号を与えるだけで済むので、デバッグ等も容易になる。

【0003】

【発明が解決しようとする課題】 しかしながら、上記構成のものでは、ウォッチドッグタイマを停止させるための信号を当該ウォッチドッグタイマの外部端子に与えなければならず、その接続が極めて面倒であった。

【0004】 本発明は上記事情に鑑みてなされたもので、その目的は、面倒な接続を行うことなく簡単に機能検査を行うことができるウォッチドッグタイマ及び斯様

なウォッチドッグタイマを備えたマイクロコンピュータを提供することにある。

【0005】

【課題を解決するための手段】 請求項 1 の発明によれば、CPU が正常に動作しているときは、CPU は、ウォッチドッグタイマに所定周期毎にアクセスするので、カウント手段は、カウント動作を繰返して実行する。この場合、ウォッチドッグタイマのカウント値が限度値となることはない。

10 【0006】 そして、CPU が暴走したときは、CPU はウォッチドッグタイマにアクセスしなくなるので、カウント手段のカウント値が限度値となる。これにより、カウント手段は異常信号を CPU へ出力するので、CPU はリセットされたり或いは割込みをかけられることにより暴走状態が停止する。

【0007】 さて、エミュレーション時においてウォッチドッグタイマの動作を検査するためにカウント手段のカウント動作を一時停止するときは、CPU によるエミュレーションプログラムの実行により停止指令発生手段に対して第 1 のアクセス条件でアクセスする。

【0008】 すると、停止指令発生手段は、カウント手段へ停止指令を出力するので、カウント手段はカウント動作を停止する。この状態でウォッチドッグタイマを動作状態を検査する。

【0009】 また、カウント手段のカウント動作を再開するときは、停止指令発生手段に対して CPU により第 2 のアクセス条件でアクセスする。すると、停止指令発生手段は、カウント手段に対する停止指令の出力を停止する。これにより、カウント手段がカウント動作を再開するようになる。

30 【0010】 従って、CPU によるエミュレーションプログラムの実行により任意のタイミングでウォッチドッグタイマを停止したり、再動作させたりすることができるので、ウォッチドッグタイマに配線を接続することなくプログラム開発のデバッグ作業ができる。

【0011】 請求項 2 の発明によれば、カウント手段のカウント動作を停止するときは、CPU により停止指令発生手段に対して第 1 のデータの書き込みを行う。すると、停止指令発生手段は、CPU により第 1 のデータの書き込み動作が行われたときは、カウント手段へ停止指令を出力する。これにより、カウント手段はカウント動作を停止するようになる。

【0012】 また、カウント手段のカウント動作を再開するときは、CPU により停止指令発生手段に対して第 2 のデータの書き込みを行う。すると、停止指令発生手段は、CPU により第 2 のデータの書き込みが行われたときは、カウント手段への停止指令の出力を停止する。これにより、カウント手段は、カウント動作を再開するようになる。

50 【0013】 従って、CPU により停止指令発生手段に

対してデータの書込み動作を実行するだけで、ウォッチドッグタイマを任意のタイミングで停止したり、再動作させたりすることができる。

【0014】請求項3の発明によれば、停止指令出力手段による停止指令の出力状態を確認するときは、外部から出力状態出力手段へ指令を与える。すると、出力状態出力手段は、停止指令発生手段による停止指令の出力状態を出力するので、その出力により停止指令の出力状態を確認することができる。

【0015】請求項4の発明によれば、カウント手段の  
10 カウント値を確認したいときは、外部からカウント値出力手段に指令を与える。すると、カウント値出力手段は、カウント手段のカウント値を出力するので、その出力によりカウント手段のカウント値を確認することができる。

【0016】

【発明の実施の形態】以下、本発明の一実施例を図面を参照して説明する。図1はワンチップマイクロコンピュータの概略構成を示すと共に、ワンチップマイクロコンピュータが有するウォッチドッグタイマの電氣的構成を  
20 示している。この図1において、ワンチップマイクロコンピュータ1は、CPU2、プログラムが記憶されたROM3、ワーキングデータ記憶用のRAM4及び周辺回路5に加えてウォッチドッグタイマ6を備えて構成されている。

【0017】CPU2とROM3、RAM4、周辺回路5及びウォッチドッグタイマ6とはアドレスバス7及びデータバス8などのシステムバスを通じて接続されており、CPU1とそれらの間でシステムバスを通じてデータの授受が行われるようになっている。

【0018】ウォッチドッグタイマ6は、カウントイネーブル発生回路9（停止指令出力手段に相当し、以下、CE発生回路と称する）、ダウンカウンタ10（カウント手段に相当）及び制御回路11を主体として構成されている。

【0019】CE発生回路9において、ラッチ回路12のデータ入力端子Dにはデータバス8のDB0（データバスのビット0）が接続されており、クロック端子Cが  
30 ハイレベルとなったタイミングでデータバスDB0のビットパターンがラッチ回路12に記憶されて出力端子Qから出力される。

【0020】ラッチ回路12のリセット端子Rには電源投入時にシステムリセット信号が入力されるようになっており、システムスタート時においてはラッチ回路12の出力端子Qの出力レベルはローレベルとなっている。

【0021】ラッチ回路12のクロック端子Cにはアンド回路13の出力端子が接続されている。このアンド回路13の入力端子にはアドレスバスデコーダ14（以下、Aバスデコーダと称する）、データバスキーパターンデコーダ15（以下、Dバスキーパターンデコーダと  
50

称する）が接続されていると共に、CPU2からのライト信号IWRE、CPU駆動用のクロック信号CK1が入力するように接続されている。

【0022】Aバスデコーダ14はアドレスバス7のビットパターンが所定パターンとなったことを検出するためのもので、例えばCE発生回路9に予め設定されたアドレスが[041C]hだとすると、アドレスデータが[041C]hとなった状態でハイレベル信号を出力する。Dバスキーパターンデコーダ15はデータバス8のビットパターンが所定パターンとなったことを検出するためのものである。

【0023】図2はDバスキーパターンデコーダ15の具体的回路を示している。この図2において、データバスDB5（ビット5）～DB1（ビット1）が所定のキーパターンとなったときにアンド回路13からの出力がハイレベルとなるように構成されている。つまり、データバスDB1、DB3、DB5はアンド回路16の入力端子に直接接続され、データバスDB2、DB4はインバータ17を介してアンド回路16の入力端子に接続されている。

【0024】従って、このような接続関係にあつては、DB5～DB1のビットパターンが[10101]bとなったときのみアンド回路16のAND条件が成立してハイレベル信号を出力し、これ以外のビットパターンではアンド回路16からハイレベル信号を出力することはない。

【0025】以上の構成により、CE発生回路9は、アドレスバス7のビットパターンが[041C]hで且つデータバス8のDB5～DB0のビットパターンが[101011]bとなった状態で（第1のアクセス条件、第1のデータの書込み動作に相当）、CPU2からライト信号IWREが入力し且つクロック信号CK1が入力したタイミングでハイレベルのカウントイネーブル信号（以下、CE信号と称する）と称する。

【0026】また、CE発生回路9は、アドレスバス7のビットパターンが[041C]hで且つデータバス8のDB5～DB0のビットパターンが[101010]bとなった状態で（第2のアクセス条件、第2のデータの書込み動作に相当）、CPU2からライト信号IWREが入力し且つクロック信号CK1が入力したタイミングでCE信号の出力を停止する。

【0027】CE発生回路9からのCE信号はバッファ回路18に与えられる。このバッファ回路18の出力端子は所定のデータバスと接続されており、データバス8を通じてCPU2がバッファ回路18の出力レベルを読  
40 取れるようになっている。

【0028】ダウンカウンタ10において、選択器19は制御回路11により決定された側のデータを入力してラッチ部20によりラッチすることによりダウンカウンタ10の値を更新するようになっている。つまり、選択

器19は、選択端子Sがローレベルの状態においては入力端子IN0からの入力データを選択して出力すると共に、選択端子Sがハイレベルの状態においては入力端子IN1の入力データを選択して出力するようになっている。この場合、選択器19の入力端子IN1はデータバス8と接続されており、選択端子SがハイレベルとなったタイミングでCPU2からのカウント初期値をデータバス8を通じてダウンカウンタ10に取込むようになっている。

【0029】ラッチ部20は、CPU駆動用のクロックCK1の入力タイミングで選択器19からの出力データをラッチして記憶すると共にその記憶データを出力する。この場合、ラッチ部20は記憶したデータが0となったときはCPU2へボロー信号（異常信号に相当）を出力するようになっている。

【0030】デクリメント部21は、ラッチ部20からの出力データを入力し、その入力データから1を減算した値を出力する。ラッチ部22は、C端子にハイレベルが入力したタイミングでデクリメント部21からの出力データを記憶すると共にその記憶データを出力する。このラッチ部22の出力端子は選択器19の入力端子IN0と接続されている。

【0031】以上の構成により、ダウンカウンタ10は、所定タイミングでCPU2によりカウント初期値が設定されると共に、所定期間毎にカウント値が1ずつ減数される。

【0032】バッファ回路23は、ダウンカウンタ10のカウント値をCPU2がリードするための回路であり、制御回路11からWDTリード信号を入力したタイミングでダウンカウンタ10からのカウント値をデータバス8を通じてCPU2へ出力する。

【0033】一方、アンド回路24は、ダウンカウンタ10がデクリメント動作を実行するためのパルス信号を発生すると共に、CE発生回路9から出力されるCE信号に応じてダウンカウンタ10に対するパルス信号の出力を禁止する機能を有する。つまり、アンド回路24の入力端子はCE発生回路9からのCE信号をインバータ25を介して入力するように接続されていると共に、CPU駆動用のクロック信号CK2及び制御回路11からのタイマクロック信号を入力するように接続されている。

【0034】従って、アンド回路24は、CE発生回路9からハイレベルのCE信号の非出力状態でクロック信号CK2が入力し且つ制御回路11からタイマクロック信号が入力したタイミングでアンド条件が成立してハイレベル信号をダウンカウンタ10へ出力する。

【0035】制御回路11は、ウォッチドッグタイマ6全体の動作を制御するためのものである。つまり、制御回路11は、CPU2により所定のアドレスにアクセスされた状態でリード信号が与えられたタイミングでカウ

ント値ライト信号をダウンカウンタ10へ出力する。また、制御回路11は、例えば1 $\mu$ s毎にタイマクロック信号をアンド回路24へ出力する機能を有する。

【0036】また、制御回路11は、自己に設定された所定のアドレスがアクセスされた状態でCPU2からリード信号が与えられたタイミングでCEリード信号を出力する。この場合、CEリード信号はバッファ回路18の制御端子に与えられるもので、その制御端子のレベルがハイレベルとなった状態でCE発生回路9からのCE信号の出力状態がバッファ回路18からデータバス8を通じてCPU2へ出力される。

【0037】また、制御回路11は、自己に設定された所定のアドレス（CEリード信号出力用のアドレスとは異なる）がアクセスされた状態でCPU2からリード信号が与えられたタイミングでWDTリード信号を出力する。この場合、WDTリード信号はバッファ回路23の制御端子に与えられるもので、その制御端子のレベルがハイレベルとなった状態でダウンカウンタ10のカウント値がバッファ回路23からデータバス8を通じてCPU2へ出力される。

【0038】さらに、制御回路11は、CPU2によりウォッチドッグタイマ10に対するカウント初期値ライトタイミングとダウンカウンタ10に対するタイマクロック信号の出力タイミングとが重なった場合のタイミング調整するためのシーケンサ機能を有する。

【0039】以上のように構成されたウォッチドッグタイマ6はCPU2の暴走を検出するためのものであり、CPU2が正常に動作している状態では、ウォッチドッグタイマ値が0となる前に当該ウォッチドッグタイマ値がCPU2により更新されるようになっている。

【0040】具体的には、更新値（カウント初期値）を例えば[C000]hとした場合、CPU2が暴走したときは、ウォッチドッグタイマ値が[0000]hになる前に更新されないこととなるので、ウォッチドッグタイマ値が0になり、ウォッチドッグタイマ6のダウンカウンタ10からボロー信号が発生してCPU2に対して暴走を知らせるために割込みをかけたり、強制的にリセットするようになる。

【0041】尚、上述のようにウォッチドッグタイマ5に設定された各アドレスはメモリ空間に配列されるものであるが、ROM3或いはRAM3のアドレスとは一致しないように設定されている。

【0042】次に上記構成の作用について説明する。ワンチップマイクロコンピュータ1を用いてプログラム開発する場合、デバッグのためのエミュレーションプログラムにより動作を検査する必要がある。この場合、エミュレーションプログラムの実行方法には、ROM3内に予め格納したのを使用したり、外部とのパソコン等とシリアル通信などを介してRAM上にエミュレーションプログラムを転送して実行させたり、ICEを用いたりす

る方法がある。

【0043】さて、CPU2は、エミュレーションプログラムの実行にしたがって、まず、ウォッチドッグタイマ6に対して[C000]hを書込む。すると、制御回路11の制御により、ダウンカウンタ10のラッチ部20には[C000]hが書込まれる。ここで、制御回路11は、タイマクロック信号を1μs毎に出力するので、ダウンカウンタ10は1μs毎に[C000]hから1ずつ減算する。

【0044】一方、CPU2は、ウォッチドッグタイマ6に対して所定周期毎に[C000]hを再書込みするので、ウォッチドッグタイマ6には所定周期毎にカウント初期値が書込まれる。この結果、ウォッチドッグタイマ6のカウント値が0となることはないので、ウォッチドッグタイマ6からボロー信号がCPU2に対して出力されてしまうことはなく、CPU2はエミュレーションプログラムの動作を継続する。

【0045】さて、CPU2は、エミュレーションプログラムの実行にしたがってウォッチドッグタイマ6の機能を検査するために当該ウォッチドッグタイマ6を停止するときは、アドレス[041C]hに[002B]hを書込む動作を実行する(図3参照)。

【0046】この結果、アドレスバス7のビットパターンが[041C]hとなると共に、データバス8においてDB5~DB0が[10101]bとなるので、Aバスデコーダ14及びDバスキーパターンデコーダ15からハイレベル信号が出力される。

【0047】そして、この状態でCPU2からライト信号IWREが出力され且つクロック信号CK1が出力されたタイミングで、アンド回路13からハイレベル信号が出力されるので、ラッチ回路12は、そのタイミングでデータバスにおけるDB0のビットパターン[1]を出力端子Qから出力する。

【0048】以上の動作により、CE発生回路9からハイレベルのCE信号が出力されるので、アンド回路24の入力端子の1つのレベルがローレベルとなり、アンド回路24からローレベル信号が出力されるようになる。従って、ダウンカウンタ10のカウント動作が停止して当該ダウンカウンタ10のカウント値が変化しなくなる(図3及び図4参照)。

【0049】上述のようにしてウォッチドッグタイマ6を停止させたCPU2は、エミュレーションプログラムの実行にしたがってウォッチドッグタイマ6の動作状態を読取る。

【0050】即ち、CPU2は、CE発生回路9からの出力状態を読取るために制御回路11に設定されたCE発生回路9に対応したアドレスを出力する。すると、制御回路11は、自己に設定されたアドレスのうちCE発生回路9に対応したアドレスが選択された状態で、CPU2からライト信号IWREを入力したときは、CEリ

ード信号をCE発生回路9に接続されたバッファ回路18に出力する。

【0051】これにより、バッファ回路18が有効となって、バッファ回路18からCE発生回路9からのCE信号の出力状態がデータバス8へ出力されるので、CPU2は、CE発生回路9のCE信号の出力状態を読取ることができる。

【0052】続いて、CPU2は、ウォッチドッグタイマ6のカウント値を読取るために制御回路11に設定されたダウンカウンタ10に対応するアドレスを出力する。すると、制御回路11は、自己に設定されたアドレスのうちダウンカウンタ10に対応したアドレスが選択された状態で、CPU2からライト信号IWREを入力したときは、WDTリード信号をダウンカウンタ10に接続されたバッファ回路23に出力する。

【0053】これにより、バッファ回路23が有効となって、バッファ回路23からダウンカウンタ10のカウント値がデータバス8へ出力されるので、CPU2は、ダウンカウンタ10の現在のカウント値を読取ることができる。

【0054】従って、CPU2が読取ったウォッチドッグタイマ6の動作状態に基づいて、エミュレーションプログラムに設定された所定のタイミングでもってマイクロコンピュータ1による動作状態を確認することができる。

【0055】上述のようにしてウォッチドッグタイマ6の動作状態を読取ったCPU2は、ウォッチドッグタイマ6を再動作させる。つまり、アドレス[041C]hに対して[002A]hを書込む動作を実行する(図3参照)。

【0056】これにより、CE発生回路9が選択されて、ラッチ回路12にはデータバス8のDB0のレベルである[0]が書込まれるので、CE発生回路9からのCE信号の出力が停止する。この結果、ダウンカウンタ10が有効となり、カウント動作を再開するようになる(図3及び図4参照)。

【0057】上記構成のものによれば、CPU2によるエミュレーションプログラムの実行によりウォッチドッグタイマ6のCE発生回路9に対してアクセスすることによりダウンカウンタ10をソフト的に停止させたり再動作させるように構成したので、ウォッチドッグタイマを停止させるための専用端子を有し、その専用端子に信号を与えることによりウォッチドッグタイマを停止させる従来構成のものと違って、面倒な接続を行うことなくウォッチドッグタイマ6を所定タイミングで簡単に停止させることができ、プログラム開発のデバッグが容易になる。さらに、プログラムの実行途中に割り込んでエミュレーションを実行させる場合にも、その間のみCE発生回路9の出力をローレベルとして、エミュレーション終了後、CE発生回路9の出力をハイレベルにすること

で、通常動作に復帰できるため、非常に使い易いものとなる。

【0058】また、ウォッチドッグタイマ6を停止させた状態で、CPU2によりウォッチドッグタイマ6のCE発生回路9の動作状態及びダウンカウンタ10のカウンタ値を読取るようにしたので、デバッグを一層容易に行うことができる。

【0059】さらに、ウォッチドッグタイマ6を停止するためのアドレス及び当該ウォッチドッグタイマ6の動作状態を読取るためのアドレスをメモリ空間に配列しながら、ROM3及びRAM4のアドレスと一致しないようにしたので、CPU2が実際のプログラムを実行した際に、ウォッチドッグタイマ6が不用意にアクセスされることがなく、ウォッチドッグタイマ6のカウント動作に支障を生じることはない。

【0060】本発明は、上記実施例にのみ限定されるものではなく、次のように変形または拡張できる。ダウンカウンタ10の代わりに、アップカウンタを設け、アップカウンタのカウント値が限度値になったときにキャリア信号をCPU2に対する異常信号として出力するようにしてもよい。

【0061】Dバスキーパターンデコーダとしては、

【10101】bというビットパターンに限定されるものではないと共に、5ビットのキーパターンに限定されるものではない。さらに、CE発生回路9用のAバスデコーダ14とは別にDバスキーパターンデコーダ15専用のアドレスを設定し、Dバスキーパターンデコーダ15が有効になった状態でのみCE発生回路9の動作を有\*

\* 効化することにより、誤動作を一層防止することができる。

【0062】また、ウォッチドッグタイマ6のアドレスをメモリ空間に配列したが、I/Oアドレス空間に配列するようにしてもよい。この場合、ウォッチドッグタイマ6にアクセスするには、CPU2によるI/O命令の実行による。

【0063】また、ウォッチドッグタイマ6を停止させた状態でその動作状態を読取る手段としては、ウォッチドッグタイマ6をCPU2から切離し、データバス8を通じてウォッチドッグタイマ6にアクセスするようにしてもよい。

【図面の簡単な説明】

【図1】本発明の一実施例におけるワンチップマイクロコンピュータの全体構成を示す概略図

【図2】Dバスキーパターンデコーダの具体的構成を示す図

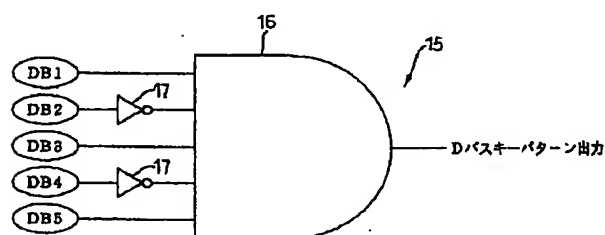
【図3】ウォッチドッグタイマの動作を示す各種信号の波形図

【図4】ウォッチドッグタイマのタイマカウント値の変化を示す図

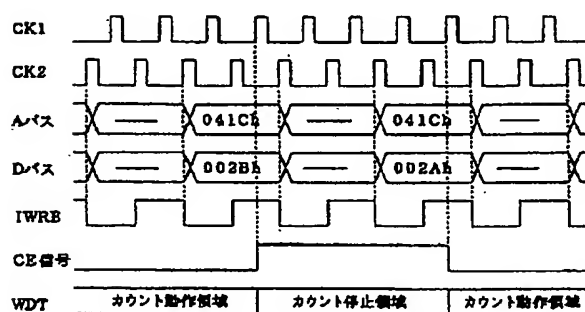
【符号の説明】

1はワンチップマイクロコンピュータ、2はCPU、6はウォッチドッグタイマ、9はカウントイネーブル発生回路（停止指令出力手段）、10はダウンカウンタ（カウント手段）、18はバッファ回路（出力状態出力手段）、23はバッファ回路（カウント値出力手段）である。

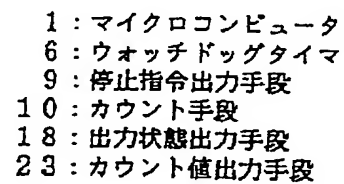
【図2】



【図3】



1





【図 4】

